基于含噪数据的深度鲁棒 Koopman 学习

Aditya Singh, Rajpal Singh, *IEEE* 研究生会员 和 Jishnu Keshavan, *IEEE* 会员

arXiv:2601.01971v1 [cs.RO] 5 Jan 2026

摘要——**Koopman 算子理论已发展为一种领先的数据驱动方法，通过审慎选择可观测函数，在提升后的可观测空间中实现非线性系统的全局线性表示。然而，真实世界的数据往往含有噪声，因此难以获得准确且无偏的 Koopman 算子近似。由含噪数据集生成的 Koopman 算子通常会受到噪声引起的偏差影响，严重降低预测和后续跟踪性能。为解决这一缺陷，本文提出一种新颖的基于自编码器的神经网络架构，从含噪数据中联合学习适当的升维函数和低偏差Koopman算子。该架构首先学习与系统正向和反向时间动力学均一致的 Koopman 基函数。随后，利用所学习的正向和反向时间动力学，合成偏差较低的 Koopman 算子，使该方法相比现有技术具有更强的抗噪鲁棒性。理论分析表明，在存在训练噪声的情况下，该方法可显著降低偏差。本文针对多个串联机械臂开展动力学预测和跟踪控制仿真，并与领先的替代设计进行性能比较，以展示该方法在各种噪声水平下的鲁棒性。此外，还通过 Franka FR3 7自由度机械臂的实验研究，进一步验证所提出方法在实际场景中的有效性。**

索引词—**Koopman 算子、噪声鲁棒性、非线性系统、数据驱动辨识、轨迹跟踪。**

I. 引言

非线性动力系统在众多科学与工程领域中一直面临着难以解决的挑战。尽管已有大量研究，但目前仍未建立起一种用于处理此类复杂系统的全面、通用的数学框架。线性控制理论为线性系统的分析与控制提供了强有力的工具，因此促使研究者致力于开发将非线性动力学纳入线性框架的方法。在这些方法中，Koopman 算子理论 [1]因能够推导出非线性系统的全局线性表示而日益受到重视。与泰勒级数近似等局部线性化技术不同，Koopman 算子理论能够在整个状态空间中以线性形式编码系统动力学，从而显著提升预测性能 [2], [3]。

Koopman理论采用一组非线性基函数，这些基函数在 Koopman 算子 [1] 的作用下，于变换后的可观测空间中呈线性演化。然而，要在可观测空间中完整表示系统动力学，通常需要无限集合的可观测函数，而这对于实际应用而言并不可行。为克服这一限制，研究人员开发了多种数据驱动方法，例如动态模态分解（DMD） [4],、带控制的动态模态分解（DMDc） [5],、扩展动态模态分解（EDMD） [6],以及非线性动力系统稀疏辨识（SINDy） [7], ，用于构建 Koopman线性模型的有限维近似。然而，上述数据驱动方法通常依赖人工选择，以确定所需的有限组可观测函数。对于较为简单的系统，先验知识可以指导这种人工选择；但对于复杂系统，辨识最优函数集会变得越来越困难且繁琐。为解决这一缺陷，研究人员提出了多种基于学习的方法，用于选择最优的可观测函数集 [8]。例如，文献 [9], 应用字典学习实现了 EDMD，而文献 [10] 将自编码器用于 SINDy，文献 [11] 则将其用于 Koopman 特征值分解。这些学习框架在生成高保真线性 Koopman 模型方面的有效性，推动了其在机械臂 [12],、四旋翼飞行器 [13], [14], 和软体机器人 [15]–[17] 等机器人系统的建模与控制中的应用。

在采用数据驱动框架准确学习 Koopman 算子时，一个重要考量是，这些框架高度依赖底层数据的性质。特别是，众所周知，从真实世界系统中获取的数据集通常含有噪声，从而导致 Koopman 算子的预测不准确 [18]。系统原始状态空间中固有的噪声通常被假设为零均值高斯噪声。然而，经过非线性升维函数变换后，其统计特性变得模糊，偏离底层高斯分布并产生非零均值。因此，高斯噪声和零均值噪声的标准假设不再成立，使得升维状态空间中的不确定性表征成为研究领域面临的开放性挑战 [19]。当底层模型未知时，对数据进行去噪变得十分困难 [20],，因为此类方法通常假设能够获得真实值，而这对于真实世界数据集往往并不成立。因此，去噪可能会扭曲底层系统动力学 [21], [22]。这凸显了设计天生具备噪声鲁棒性的Koopman架构的必要性，从而确保其在真实世界场景中的预测性能可靠。因此，近年来已有多种 Koopman架构被提出，用于处理含噪数据。特别地，文献 [18]，研究了噪声对 DMD 的影响，并在同一研究中随后提出了包括前向-后向 DMD 在内的多种方法，以校正噪声引起的偏差。然而，由于 DMD 仅将系统的物理状态用作可观测量，零均值高斯噪声模型仍然有效，但这使其无法捕捉

本研究部分得到科学与工程研究委员会（SERB）CoreResearchGrant 的资助。作者均就职于印度班加罗尔卡纳塔克邦 560012 印度科学理工学院机械工程系（电子邮箱：adityasingh2@iisc.ac.in、rajpalsingh@iisc.ac.in、kjishnu@iisc.ac.in）。

高度非线性系统的动力学。因此，文献 [23] 提出了带输入的前向-后向 EDMD，对该架构进行了扩展，但仍然假设升维状态中的噪声为零均值噪声。尽管如此，如前文所述，在不使用深度学习架构的情况下，为非线性系统寻找合适的 Koopman 可观测函数仍然是一项挑战。为克服这一缺点，文献 [24]，提出了基于噪声数据的深度 Koopman 学习（DKND）方案，通过调整基函数来减小噪声的影响。然而，该方法忽略了在含噪观测下通过线性回归估计 Koopman 算子时产生的显著偏差。因此，学习得到的算子仍保留了较大的偏差，导致预测精度降低。文献 [25] 提出了另一种基于深度学习的架构，通过增加展开时长来最小化循环损失；该方法依赖于逐步增加用于计算预测损失的时间范围。然而，该研究既未考察噪声的影响，也未提供鲁棒性的数学保证，并且带来了显著的计算负担。文献 [26]，提出了一种自适应 Koopman 框架，通过在线学习修改标称 Koopman 模型，以适应测量值和系统动力学中存在的不确定性。然而，该研究使用从仿真中获取的无噪声数据离线学习标称模型，未针对噪声鲁棒性提供任何明确保证。最后，文献 [27] 利用自编码器，同时借助自主系统的前向和后向动力学，生成一致的 Koopman 可观测量。与本文研究不同，该方法未考察噪声的影响，因此未对所得 Koopman 算子的噪声鲁棒性提供理论保证。此外，该方法仅适用于自治系统，无法扩展到控制仿射系统。

为克服这些挑战，我们提出深度鲁棒 Koopman 网络（DRKN）框架，以减轻训练数据中的噪声所引起的偏差。该方法基于 [18], 中的前向-后向 DMD 概念，并将分析扩展到具有非线性 Koopman 可观测量的控制仿射系统。通过将这一概念融入基于自编码器的神经网络架构，DRKN 能够实现最优基函数的数据驱动发现，同时学习与控制仿射系统前向和后向时间动力学相对应的 Koopman 算子。提出的方法利用前向–后向动力系统的解析特性，更准确地（以更低偏差）估计 Koopman 算子。因此，无需预先了解提升空间中噪声的统计特性，就能增强对含噪数据的鲁棒性，进而为高度非线性系统辨识出可靠模型。据作者所知，这是首项明确刻画在升维状态空间中使用非线性基函数学习 Koopman 算子时噪声引起的偏差的研究；在该空间中，噪声不再是零均值高斯噪声。本文通过理论分析，在对观测噪声统计特性作出温和假设的条件下，为所提出的方法提供鲁棒性保证。全面的仿真和实验研究（包括与领先替代设计的比较）用于展示所提出方案的有效性和优越性能。

因此，本文的主要贡献如下：

1）我们提出了一种新颖的神经网络架构，通过前向-后向动力学传播，从含噪数据中为高度非线性的控制仿射系统学习精确的 Koopman 算子。

2）与依赖人工选择基函数的既有前向-后向 Koopman 框架不同，所提出的架构采用自编码器自动学习最优基函数集。3）我们对噪声鲁棒性进行了理论分析，明确证明了采用所提出方案综合 Koopman 算子时的偏差降低。4）通过对多个串联机械臂系统进行闭环轨迹跟踪控制，展示了所提出方案的有效性。特别是，通过在 Franka FR3 七自由度系统上开展跟踪控制实验，清晰展示了所提出方法的可扩展性；相较于当前最先进的方法，这是其一项显著优势。

本文其余部分安排如下。第二节首先介绍 Koopman 算子理论。第三节概述所提出的深度鲁棒 Koopman 网络（DRKN）架构。第四节提供理论分析，包括关键假设和相关证明。第五节通过仿真研究展示该架构的噪声鲁棒性。第六节展示采用所提出架构开展的真实场景实验。最后，第七节给出结论。

II. 预备知识

A. 背景

为奠定基础，我们首先简要介绍 Koopman 算子理论 [1]。考虑如下非线性自主系统：

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}), \tag{1}$$

其中，系统的状态 $\boldsymbol{x} \in \mathbb{X} \subset \mathbb{R}^n$ 和 $\boldsymbol{f}\colon\ \mathbb{X} \to \mathbb{X}$ 假设在 $\mathbb{X}$ 上是 Lipschitz 连续的。相应的离散动力学可表示为 $\boldsymbol{x}_{k+1} = \boldsymbol{F}(\boldsymbol{x}_k)$，其中 $\boldsymbol{x}_k$ 描述第 k^{th} 个时间步的状态，$\boldsymbol{S}\colon\ \mathbb{X} \to \mathbb{X}$ 表示流映射。

一组位于希尔伯特空间中的可观测函数 $\sigma\colon\ \mathbb{X} \to \mathbb{C}$ 的演化由连续 Koopman 算子 $\mathcal{K}\colon\ \mathbb{C} \to \mathbb{C}$ 以线性方式描述，使得

$$\mathcal{K} \circ \sigma(\boldsymbol{x}_k) = \sigma \circ \boldsymbol{S}(\boldsymbol{x}_k) = \sigma(\boldsymbol{x}_{k+1}),$$

其中，$\circ$ 表示函数复合。请注意，可观测量在 Koopman 空间中的演化是线性的。然而，精确的 Koopman表示需要是无限维的。因此，在实际应用中通常使用Koopman 算子的有限维近似。

相同的理论概念也可以扩展到带有控制输入的系统。考虑如下形式的控制仿射系统：

$$\dot{\boldsymbol{x}} = \boldsymbol{f}_0(\boldsymbol{x}) + \sum_{i=1}^{m} \boldsymbol{f}_i(\boldsymbol{x}) u_i, \tag{2}$$

其中，$\boldsymbol{u} = [u_1, u_2, ..., u_m]^\top \in \mathbb{R}^m$ 表示系统的控制输入，$\boldsymbol{f}_0$ 和 $\boldsymbol{f}_i$ 分别表示漂移项和状态依赖的控制向量。然后，对于$\boldsymbol{z} = [\phi_1(\boldsymbol{x})\ \phi_2(\boldsymbol{x})\ ...,\ \phi_N(\boldsymbol{x})]^T$的选择，相应的Koopman表示为：

$$\dot{\boldsymbol{z}} = \boldsymbol{A_c} \boldsymbol{z} + \boldsymbol{B_c} \boldsymbol{u}, \boldsymbol{x} = \boldsymbol{C} \boldsymbol{z}, \tag{3}$$

其中，$\phi_i(\cdot): \mathbb{X} \to \mathbb{C}\ \forall\ i = 1, 2, ..., N$表示用于构造Koopman 算子有限维近似的可观测函数，$\boldsymbol{A_c} \in \mathbb{R}^{N \times N}$和 $\boldsymbol{B_c} \in \mathbb{R}^{N \times m}$ 表示相应的Koopman矩阵。$\boldsymbol{C} \in \mathbb{R}^{n \times N}$ 表示从提升空间状态 $\boldsymbol{z}$ 到系统基状态 $\boldsymbol{x}$ 的线性映射。线性Koopman动力学对应的离散时间表示可写为

$$\boldsymbol{z}_{k+1} = \boldsymbol{A} \boldsymbol{z}_k + \boldsymbol{B} \boldsymbol{u}_k,\ \boldsymbol{x}_k = \boldsymbol{C} \boldsymbol{z}_k. \tag{4}$$

关于控制仿射系统Koopman 算子的更深入分析可见 [26], [28]。

B. 问题表述

考虑无噪声数据集 $\mathcal{D}_a = (\{\boldsymbol{x_i}\}_{i=1}^{N_s},\ \{\boldsymbol{x'_i}\}_{i=1}^{N_s},\ \{\boldsymbol{u_i}\}_{i=1}^{N_s})$，其表示由式（2）定义的系统动力学，其中 $\{\boldsymbol{x'_i}\}_{i=1}^{N_s}$ 表示在状态 $\{\boldsymbol{x_i}\}_{i=1}^{N_s}$ 下向系统施加相应控制输入 $\{\boldsymbol{u_i}\}_{i=1}^{N_s}$ 后得到的单步演化状态。采集到的相关含噪测量值收集于 $\mathcal{D}_m = (\{\boldsymbol{x_{mi}}\}_{i=1}^{N_s},\ \{\boldsymbol{x'_{mi}}\}_{i=1}^{N_s},\ \{\boldsymbol{u_{mi}}\}_{i=1}^{N_s})$ 中。对于每个含噪状态 $\boldsymbol{x_{mi}}$，其提升表示定义为 $\boldsymbol{z_{mi}} = \boldsymbol{\phi}(\boldsymbol{x_{mi}})$，其中 $\boldsymbol{\phi} = [\phi_1, \phi_2, \ldots, \phi_N]^\top$ 表示用于 Koopman 升维的可观测函数字典。

现在，利用可用的含噪数据对 Koopman 算子 $\boldsymbol{K_{fm}}$ 进行近似，可得：

$$\boldsymbol{K_{fm}} = \arg\min_{\boldsymbol{K_m}} \left\| \begin{bmatrix} \boldsymbol{Z'_m} \\ \boldsymbol{U_m} \end{bmatrix} - \boldsymbol{K_m} \begin{bmatrix} \boldsymbol{Z_m} \\ \boldsymbol{U_m} \end{bmatrix} \right\|_F, \tag{5}$$

其中，$\boldsymbol{Z_m}$ $\{\boldsymbol{z_{mi}} = \boldsymbol{\phi}\ \boldsymbol{x_{mi}}\}_{i=1}^{N_s}$ 是提升状态()的快照矩阵，$\boldsymbol{Z'_m}$ 而是下一时间步的单步前向提升状态的快照矩阵 $\{\boldsymbol{z'_{mi}} = \boldsymbol{\phi}(\boldsymbol{x'_{mi}})\}_{i=1}^{N_s}$，对应于控制输入 $\boldsymbol{u_m}$ 作用于 $\boldsymbol{Z_m}$ 时的状态。$\boldsymbol{U_m}$是一个以控制输入为列、对应于 $\boldsymbol{Z_m}$中状态的矩阵。

另一方面，使用无噪声数据计算得到的 Koopman 算子为：

$$\boldsymbol{K_f} = \arg\min_{\boldsymbol{K}} \left\| \begin{bmatrix} \boldsymbol{Z'} \\ \boldsymbol{U} \end{bmatrix} - \boldsymbol{K} \begin{bmatrix} \boldsymbol{Z} \\ \boldsymbol{U} \end{bmatrix} \right\|_F, \tag{6}$$

其中，$\boldsymbol{Z}$、$\boldsymbol{Z'}$ 和 $\boldsymbol{U}$ 是分别对应于$\boldsymbol{Z_m}$、$\boldsymbol{Z'_m}$ 和 $\boldsymbol{U_m}$的无噪声矩阵。

将线性最小二乘求逆作为系统（5）的解，会导致在含噪数据下对 Koopman 算子 $\boldsymbol{K_{fm}}$ 产生有偏估计，从而不同于真实无噪声 Koopman算子 $\boldsymbol{K_f}$。在实践中，无噪声数据通常不可获得，这给从含噪数据中准确综合真实Koopman算子带来了挑战。噪声滤波未必总是合适，因为它可能引入失真，导致数据表现为改变后的动力学，而非真实的底层系统 [20], [21]。为克服这些挑战，本研究旨在提出一种抗噪 Koopman架构，用于从含噪训练数据中综合低偏差 Koopman算子。

III. 深度鲁棒 Koopman 网络

本节提出一种抗噪声神经网络架构，通过利用前向和后向时间系统动力学，估计有限维低偏差 Koopman 算子。

首先，我们定义与前向和后向动力学相关的关键术语如下：

$$\boldsymbol{z}_{k+1} = \boldsymbol{A_f} \boldsymbol{z}_k + \boldsymbol{B_f} \boldsymbol{u}_k, \boldsymbol{z}_{k-1} = \boldsymbol{A_b} \boldsymbol{z}_k + \boldsymbol{B_b} \boldsymbol{u}_{k-1}, \boldsymbol{x}_k = \boldsymbol{C} \boldsymbol{z}_k, \tag{7}$$

其中，$\{\boldsymbol{A_f}, \boldsymbol{B_f}\}$ 和 $\{\boldsymbol{A_b}, \boldsymbol{B_b}\}$ 分别表示正向和反向时间动力学的 Koopman 模型矩阵。

所提出的深度鲁棒 Koopman 网络（DRKN）的架构如图 1 所示。所提出的架构包含一个用于将状态映射到可观测空间的自编码器，由两个结构相同的全连接线性层组成，用于学习系统的正向和反向时间动力学。该结构确保自编码器学得的 Koopman 可观测量在正向和反向时间动力学中保持一致，这反映了一个事实：张成 Koopman 子空间的特征函数不受时间流方向的影响。因此，正向时间动力学的 Koopman 算子的有限维表示是反向时间动力学对应表示的逆矩阵 [29]。在本研究中，将系统的原始状态 $\boldsymbol{x_k}$ 与自编码器学得的非线性可观测量 $\tilde{\boldsymbol{z}}_{\boldsymbol{k}}$ 沿垂直方向堆叠，得到提升状态 $\boldsymbol{z_k} = [\boldsymbol{x_k}, \tilde{\boldsymbol{z}}_{\boldsymbol{k}}]^\top$，再通过线性层传播，以学习正向和反向时间动力学。解码器 $\boldsymbol{C} = [\boldsymbol{I}_{n \times n}\ \boldsymbol{0}] \in \mathbb{R}^{n \times N}$ 用于将提升状态投影回原始状态。该架构使神经网络能够通过最小化由以下各项组成的损失函数，同时学习基函数和 Koopman 算子：

1)前向预测损失 - 该项表示前向时间动力学在下一时间步的预测状态与实际状态之间的差异。预测的提升状态使用全连接前向线性层当前的权重进行估计，然后投影得到预测的原始状态。其表示为 $L_{fpred} = \|\boldsymbol{x_{k+1}} - \hat{\boldsymbol{x}}_{\boldsymbol{k+1}}\| = \|\boldsymbol{x_{k+1}} - \boldsymbol{C}(\boldsymbol{A_{fm}} \boldsymbol{\zeta}(\boldsymbol{x_k}, \boldsymbol{W_E}) + \boldsymbol{B_{fm}} \boldsymbol{u_k})\|$，其中$\boldsymbol{A_{fm}} \in \mathbb{R}^{N \times N}$ 和 $\boldsymbol{B_{fm}} \in \mathbb{R}^{N \times m}$ 是全连接前向线性层权重的矩阵表示，如图 1 所示。

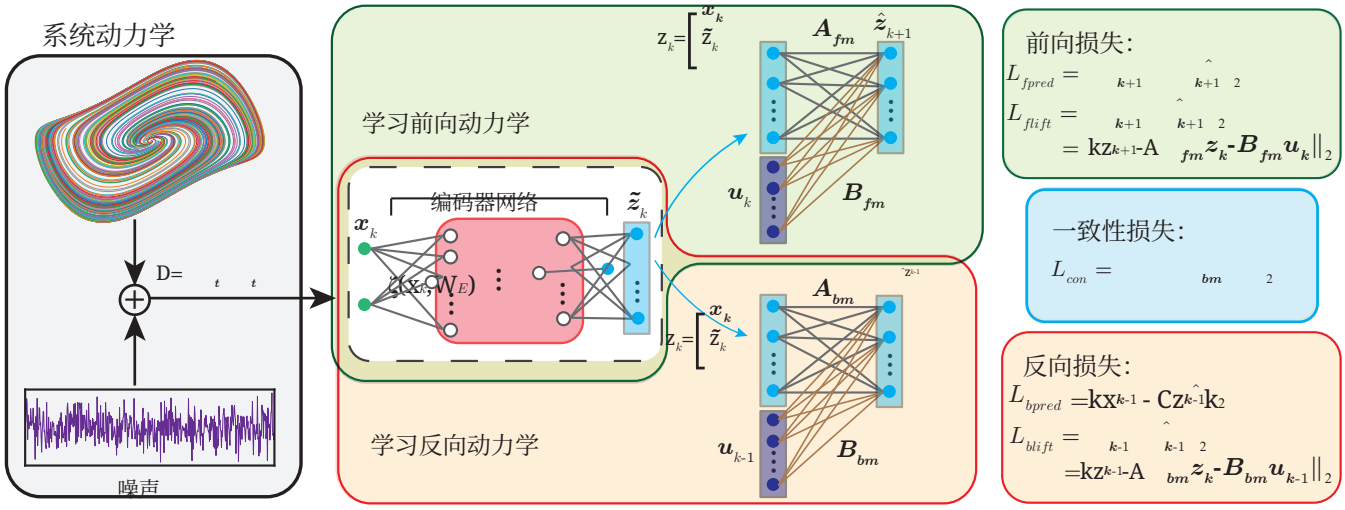

图 1. 提出的 DRKN架构示意图

2) 前向提升状态预测损失 - 该项表示前向时间动力学在下一时间步的预测提升观测量与实际观测量之间的差异。其表示为 $L_{flift} = \|z_{k+1} - \hat{z}_{k+1}\| = \|\zeta(x_{k+1}, W_E) - A_{fm}\zeta(x_k, W_E) - B_{fm}u_k\|$。3) 反向预测损失 - 该项量化了反向时间动力学在前一时间步的实际状态与预测状态之间的差异。其表示为 $L_{bpred} = \|x_{k-1} - \hat{x}_{k-1}\| = \|x_{k-1} - C(A_{bm}\zeta(x_k, W_E) + B_{bm}u_{k-1})\|$，其中 $A_{bm} \in \mathbb{R}^{N\times N}$ 和 $B_{bm} \in \mathbb{R}^{N\times m}$ 是全连接反向线性层权重的矩阵表示，如图 1 所示。4) 反向提升预测损失 - 该项表示反向时间动力学在前一时间步的实际提升观测量与预测提升观测量之间的差异。其定义为 $L_{blift} = \|z_{k-1} - \hat{z}_{k-1}\| = \|\zeta(x_{k-1}, W_E) - A_{bm}\zeta(x_k, W_E) - B_{bm}u_{k-1}\|$。5) 一致性损失 - 一致性意味着针对同一组基函数，学习到的前向和反向 Koopman 算子互为逆算子。该项通过损失项 $L_{con} = ||K_{fm}K_{bm} - I||$ 量化，其中 K_{fm} 和 K_{bm} 分别由式 (8) 和式 (9) 定义。该损失项的唯一目的是促进学习得到的基函数同时与前向和反向动态保持一致。

总体而言，目标是在整个数据集上最小化总损失，$L(x_k, x_{k+1}, x_{k-1}, u_k) = \alpha_1(L_{fpred} + L_{bpred}) + \alpha_2(L_{flift} + L_{blift}) + \alpha_3 L_{con} + \gamma_1\|W_E\|_1 + \gamma_2\|W_E\|_2$，其中 α_1、α_2、α_3、α_4、γ_1、γ_2 为标量权重。该架构利用正向和反向流的这一特性，从含噪数据中获得抗噪 Koopman 算子。

从含噪测量数据中学习得到的前向 Koopman 算子呈分块上三角形式：

$$K_{fm} = \arg\min_{K_f} \left\| \begin{bmatrix} Z'_m \\ U_m \end{bmatrix} - K_f \begin{bmatrix} Z_m \\ U_m \end{bmatrix} \right\|_F = \begin{bmatrix} A_{fm} & B_{fm} \\ 0 & I \end{bmatrix}. \quad (8)$$

算法 1 DRKN Koopman 算子综合

要求：数据集 D $= (X_t, U_t)_{t=1}^T$，其中 T 为轨迹数量。 1: 初始化神经网络参数 $\zeta(\cdot)$、A_{fm}、B_{fm}、A_{bm} 和 B_{bm}。 2: 对于每个训练轮次 i，从 1 到 E，执行 3: 打乱数据的批次顺序 4: 对于每个批次，执行 5: 1）将批次数据输入，得到提升状态 z_k、z_{k-1} 和 z_{k+1}。 6: 2）使用公式（7）计算前向和后向预测状态 $\hat{z}_{k+1}$ 和 $\hat{z}_{k-1}$。 7: 3）计算总损失 $L(x_k, x_{k+1}, x_{k-1}, u_k)$。 8: 4）优化 $\zeta(.)$、A_{fm}、B_{fm}、A_{bm} 和 B_{bm}。 9: 结束循环 10: 结束循环 11: 返回 A_{fm}、B_{fm}、A_{bm}、B_{bm}12: 使用公式（10）得到减偏算子 $K_{proposed}$。

同理，可得反向 Koopman 算子：

$$K_{bm} = \arg\min_{K_b} \left\| \begin{bmatrix} Z_m \\ U_m \end{bmatrix} - K_b \begin{bmatrix} Z'_m \\ U_m \end{bmatrix} \right\|_F = \begin{bmatrix} A_{bm} & B_{bm} \\ 0 & I \end{bmatrix}. \quad (9)$$

于是，降阶偏置前向 Koopman 算子可写为：

$$K_{proposed} = \sqrt{K_{fm}K_{bm}^{-1}} = \begin{bmatrix} A_p & B_p \\ 0 & I \end{bmatrix}. \quad (10)$$

注意，这个降阶偏置 Koopman 算子保留了块上三角形式，与式 (8) 和 (9) 中的前向和后向动力学一致。

IV. 理论分析

本节将对使用所提出方案得到的 Koopman 算子的噪声鲁棒性进行理论分析。考虑随机选取的有限状态向量序列 $\{x_i\}_{i=1}^s$，以及有限序列

的控制输入 $\{\boldsymbol{u_i}\}_{i=1}^{s}$ 作用于系统 (2)。因此，我们得到相关的状态序列 $\{\boldsymbol{x_i^+}\}_{i=1}^{s}$ 和 $\{\boldsymbol{x_i^-}\}_{i=1}^{s}$，其关系如下：

$$\boldsymbol{x_i} = \boldsymbol{f_d}(\boldsymbol{x_i^-}, \boldsymbol{u_i}), \boldsymbol{x_i^+} = \boldsymbol{f_d}(\boldsymbol{x_i}, \boldsymbol{u_i}), \tag{11}$$

其中 $\boldsymbol{f_d}$ 是（2）的离散时间表示。

令 $\{\boldsymbol{n_{xi}}\}_{i=1}^{s}$、$\{\boldsymbol{n_{xi}^+}\}_{i=1}^{s}$、$\{\boldsymbol{n_{xi}^-}\}_{i=1}^{s}$ 和 $\{\boldsymbol{n_{ui}}\}_{i=1}^{s}$ 所表示的序列分别定义序列 $\{\boldsymbol{x_i}\}_{i=1}^{s}$、$\{\boldsymbol{x_i^+}\}_{i=1}^{s}$ $\{\boldsymbol{x_i^-}\}_{i=1}^{s}$ $\{\boldsymbol{u_i}\}_{i=1}^{s}$ 和 $\{\boldsymbol{x_{mi}}\}_{i=1}^{s} = \{\boldsymbol{x_i}\}_{i=1}^{s} + \{\boldsymbol{n_{xi}}\}_{i=1}^{s}$ $\{\boldsymbol{x_{mi}^+}\}_{i=1}^{s} = \{\boldsymbol{x_i^+}\}_{i=1}^{s} +$ 中 $\{\boldsymbol{n_{xi}^+}\}_{i=1}^{s}$ $\{\boldsymbol{x_{mi}^-}\}_{i=1}^{s} = \{\boldsymbol{x_i^-}\}_{i=1}^{s} + \{\boldsymbol{n_{xi}^-}\}_{i=1}^{s}$ $\{\boldsymbol{u_{mi}}\}_{i=1}^{s} =$ 相应状态测量值的噪声分量，即,,$\{\boldsymbol{u_i}\}_{i=1}^{s} + \{\boldsymbol{n_{ui}}\}_{i=1}^{s}$。注意，假设噪声服从零均值的正态分布（独立同分布）。现在，可以将每个序列的元素水平堆叠，以生成下面定义的相应矩阵。

$$\boldsymbol{X} = [\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_s}],\ \boldsymbol{X^+} = [\boldsymbol{x_1^+}, \boldsymbol{x_2^+}, \ldots, \boldsymbol{x_s^+}],$$
$$\boldsymbol{X^-} = [\boldsymbol{x_1^-}, \boldsymbol{x_2^-}, \ldots, \boldsymbol{x_s^-}],\ \boldsymbol{U} = [\boldsymbol{u_1}, \boldsymbol{u_2}, \ldots, \boldsymbol{u_s}],$$

其中，$\boldsymbol{X}$、$\boldsymbol{X^+}$、$\boldsymbol{X^-} \in \mathbb{R}^{n\times s}$ 和 $\boldsymbol{U} \in \mathbb{R}^{m\times s}$ 分别表示对应于真实系统状态和输入矩阵的矩阵。矩阵 $\boldsymbol{N_x} = (\boldsymbol{n_{xi}})_{i=1}^{s}$、$\boldsymbol{N_x^+} = \boldsymbol{n_{xi}^+}\ _{i=1}^{s}$ $\boldsymbol{N_x^-} = \boldsymbol{n_{xi}^-}\ _{i=1}^{s} \in \mathbb{R}^{n\times s}$ ( )，( ) 表示与这些状态矩阵相关的噪声分量。因此，测得的（含噪）状态快照矩阵表示为 $\boldsymbol{X_m} = (\boldsymbol{x_{mi}})_{i=1}^{s} = \boldsymbol{X} + \boldsymbol{N_x}$、$\boldsymbol{X_m^+} = (\boldsymbol{x_{mi}^+})_{i=1}^{s} = \boldsymbol{X^+} + \boldsymbol{N_x^+}$、$\boldsymbol{X_m^-} = (\boldsymbol{x_{mi}^-})_{i=1}^{s} = \boldsymbol{X^-} + \boldsymbol{N_x^-}$，测量输入快照矩阵表示为 $\boldsymbol{U_m} = \boldsymbol{U} + \boldsymbol{N_u}$。测得状态的提升状态矩阵记为 $\boldsymbol{Z_m} = \boldsymbol{Z} + \boldsymbol{N_z}$，其中 $\boldsymbol{Z} \in \mathbb{R}^{N\times s}$ 对应真实状态的提升表示，而 $\boldsymbol{N_z} = [\boldsymbol{\phi(x_{m1})} - \boldsymbol{\phi(x_1)}\ \boldsymbol{\phi(x_{m2})} - \boldsymbol{\phi(x_2)} \ldots, \boldsymbol{\phi(x_{ms})} - \boldsymbol{\phi(x_s)}] \in \mathbb{R}^{N\times s}$则表示提升空间中的相关噪声。类似地，对应于 $\boldsymbol{Z^+}$和 $\boldsymbol{Z^-}$ 的提升测量快照矩阵分别记为 $\boldsymbol{Z_m^+}$ 和 $\boldsymbol{Z_m^-}$，其中 $\boldsymbol{N_z^+}$和 $\boldsymbol{N_z^-}$表示提升空间中的相关噪声。

现在，通过直接使用含噪测量数据求解以下优化问题，可以得到有偏的前向 Koopman 算子 $\boldsymbol{K_{fm}}$。

$$\boldsymbol{K_{fm}} = \arg\min_{\boldsymbol{K_m}} ||\boldsymbol{\Psi_m^+} - \boldsymbol{K_m}\boldsymbol{\Psi_m}||_F$$
$$\boldsymbol{\Psi_m^+} = \begin{bmatrix}\boldsymbol{Z_m^+}\\ \boldsymbol{U_m}\end{bmatrix},\ \boldsymbol{\Psi_m} = \begin{bmatrix}\boldsymbol{Z_m}\\ \boldsymbol{U_m}\end{bmatrix},\ \boldsymbol{\Psi_m^-} = \begin{bmatrix}\boldsymbol{Z_m^-}\\ \boldsymbol{U_m}\end{bmatrix}. \tag{12}$$

类似地，通过求解以下优化问题，可以得到反向 Koopman 算子 $\boldsymbol{K_{bm}}$：

$$\boldsymbol{K_{bm}} = \arg\min_{\boldsymbol{K_m}} ||\boldsymbol{\Psi_m^-} - \boldsymbol{K_m}\boldsymbol{\Psi_m}||_F, \tag{13}$$

其中，$\boldsymbol{\Psi_m^-}$、$\boldsymbol{\Psi_m}$、$\boldsymbol{\Psi_m^+} \in \mathbb{R}^{(N+m)\times s}$可进一步展开为 $\boldsymbol{\Psi_m^-} = \boldsymbol{\Psi^-} + \boldsymbol{N_\psi^-}$、$\boldsymbol{\Psi_m} = \boldsymbol{\Psi} + \boldsymbol{N_\psi}$和 $\boldsymbol{\Psi_m^+} = \boldsymbol{\Psi^+} + \boldsymbol{N_\psi^+}$。为进行进一步分析，现考虑以下假设：

假设 *1*- [26]: 假定通过自编码器学习得到的有限组基函数$\{\phi_1, ..., \phi_N\}$与输入 $\{u_1, ..., u_s\}$ 共同张成 *Koopman* 不变子空间。

假设 *2*- [18]: 假定系统动力学在时间上可逆，即 $\boldsymbol{K_b} = \boldsymbol{K_f^{-1}}$ $\boldsymbol{K_f}$ = 存在，其中 $\boldsymbol{\Psi^+\Psi^\top}\ \boldsymbol{\Psi\Psi^\top}^{-1}$（）是针对底层无噪声系统计算得到的真实 *Koopman* 算子。

假设 *3*— [30] 范数 $||\boldsymbol{K_f}|| \approx ||\boldsymbol{K_b}|| \approx \boldsymbol{1}$。

假设*4*——噪声水平足够小，使得以下不等式成立，

$$||(\boldsymbol{\Psi_i}\boldsymbol{\Psi_j^\top})^{-1}||\,||\boldsymbol{N_{\psi_i}}\boldsymbol{\Psi_i^\top} + \boldsymbol{\Psi_k}\boldsymbol{N_{\psi_j}^\top} + \boldsymbol{N_{\psi_i}}\boldsymbol{N_{\psi_j}^\top}|| \ll \boldsymbol{1}, \tag{14}$$

$\forall \boldsymbol{\Psi_i}\,\boldsymbol{\Psi_j}\,\boldsymbol{\Psi_k} \in \boldsymbol{\Psi^-}\,\boldsymbol{\Psi^+}$，$\boldsymbol{N_{\psi_i}}\boldsymbol{N_{\psi_j}} \in \boldsymbol{N_\psi^-}\ \boldsymbol{N_\psi}\boldsymbol{N_\psi^+}$。

备注 *1*：假设 1 对线性 Koopman 模型的综合提出了一个温和条件，与 [26]和 [12]中考虑的条件类似。假设 2 认为系统动力学在时间上可逆，这对大多数物理系统而言都成立 [18]。假设 3 涉及 Koopman 算子的范数；如 [30]所示，对于保测的保守动力系统，该条件已知成立。在含有控制输入的系统中，算子范数可能略微超过 1。然而，该假设的主要目的是确定 Koopman 算子范数的近似量级，从而便于后续分析。假设 4 涉及提升空间中噪声水平与系统状态的相对大小。由于自编码器采用有界激活函数的结构，加之自编码器网络的输出层学习到的权重有限，因此提升函数 $\boldsymbol{\phi}$是全局利普希茨连续且有界的 [31]。在这些条件下，矩阵$\boldsymbol{N_\psi}$，$\boldsymbol{N_\psi^+}$和 $\boldsymbol{N_\psi^-}$的范数大致处于同一量级；这些矩阵包含提升后的噪声分量及其经过一个时间步演化后的提升噪声分量，即 $||\boldsymbol{N_\psi}|| \approx ||\boldsymbol{N_\psi^+}|| \approx ||\boldsymbol{N_\psi^-}||$。这确保对于噪声矩阵和状态矩阵的所有允许组合，假设 4 中给出的不等式均成立。

现在，我们研究噪声对 Koopman 算子学习的影响。所提出的基于自编码器的神经架构通过最小化适当的损失函数，促进自动选择 Koopman 基函数。因此，一旦自编码器收敛且升维函数学习完成，后续的线性层便通过有效求解式 (12) 中的优化问题来确定 Koopman 算子。

对于 $\boldsymbol{\Psi_m} \in \mathbb{R}^{(N+m)\times s}$ 且 $s > (N+m)$，可通过调用摩尔—彭若斯伪逆 $\boldsymbol{\Psi_m}^\dagger = \boldsymbol{\Psi_m^\top}(\boldsymbol{\Psi_m}\boldsymbol{\Psi_m^\top})^{-1}$ 得到式 (12) 的解：

$$\begin{aligned}\boldsymbol{K_{fm}} &= \boldsymbol{\Psi_m^+}\boldsymbol{\Psi_m}^\dagger = (\boldsymbol{\Psi^+} + \boldsymbol{N_\psi^+})(\boldsymbol{\Psi} + \boldsymbol{N_\psi})^\dagger,\\ &= (\boldsymbol{\Psi^+} + \boldsymbol{N_\psi^+})(\boldsymbol{\Psi} + \boldsymbol{N_\psi})^\top \big((\boldsymbol{\Psi} + \boldsymbol{N_\psi})(\boldsymbol{\Psi} + \boldsymbol{N_\psi})^\top\big)^{-1},\\ &= (\boldsymbol{\Psi^+\Psi^\top} + \boldsymbol{N_\psi^+\Psi^\top} + \boldsymbol{\Psi^+N_\psi^\top} + \boldsymbol{N_\psi^+N_\psi^\top})(\boldsymbol{\Psi\Psi^\top} +\\ &\quad \boldsymbol{N_\psi\Psi^\top} + \boldsymbol{\Psi N_\psi^\top} + \boldsymbol{N_\psi N_\psi^\top})^{-1}.\end{aligned} \tag{15}$$

对于每个满足 $||\boldsymbol{M^{-1}P}|| \ll 1$ 的 $p \in \mathbb{N}$和 $\boldsymbol{M}$，$\boldsymbol{P} \in \mathbb{R}^{p\times p}$，由 [18], [32]可得

$$(\boldsymbol{M} + \boldsymbol{P})^{-1} \approx \boldsymbol{M^{-1}} - \boldsymbol{M^{-1}PM^{-1}}. \tag{16}$$

然后，结合 $||\boldsymbol{M}^{-1}\boldsymbol{P}|| \lhd ||\boldsymbol{M}^{-1}||\,||\boldsymbol{P}||$ 这一事实应用 (14)，并利用 (16)，我们得到：

$$\begin{aligned}\boldsymbol{K_{fm}}=&(\boldsymbol{\Psi}^{+}\boldsymbol{\Psi}^{\top}+\boldsymbol{N_{\psi}^{+}}\boldsymbol{\Psi}^{\top}+\boldsymbol{\Psi}^{+}\boldsymbol{N_{\psi}^{\top}}+\boldsymbol{N_{\psi}^{+}}\boldsymbol{N_{\psi}^{\top}})\\&\times(\boldsymbol{\Psi}\boldsymbol{\Psi}^{\top})^{-1}(\boldsymbol{I}-\boldsymbol{P_1})=\underbrace{\boldsymbol{K_f}(\boldsymbol{I}-\boldsymbol{P_1})}_{T_1}\\&+\underbrace{(\boldsymbol{N_{\psi}^{+}}\boldsymbol{\Psi}^{\top}+\boldsymbol{\Psi}^{+}\boldsymbol{N_{\psi}^{\top}}+\boldsymbol{N_{\psi}^{+}}\boldsymbol{N_{\psi}^{\top}})(\boldsymbol{\Psi}\boldsymbol{\Psi}^{\top})^{-1}}_{\boldsymbol{T_2}}\\&-\underbrace{(\boldsymbol{N_{\psi}^{+}}\boldsymbol{\Psi}+\boldsymbol{\Psi}^{+}\boldsymbol{N_{\psi}^{\top}}+\boldsymbol{N_{\psi}^{+}}\boldsymbol{N_{\psi}^{\top}})(\boldsymbol{\Psi}\boldsymbol{\Psi}^{\top})^{-1}\boldsymbol{P_1}}_{\boldsymbol{T_3}}.\end{aligned}\tag{17}$$

其中，$\boldsymbol{P_1}=(\boldsymbol{N_{\psi}}\boldsymbol{\Psi}^{\top}+\boldsymbol{\Psi}\boldsymbol{N_{\psi}^{\top}}+\boldsymbol{N_{\psi}}\boldsymbol{N_{\psi}^{\top}})(\boldsymbol{\Psi}\boldsymbol{\Psi}^{\top})^{-1}$，且 $\boldsymbol{K_f}=\boldsymbol{\Psi}^{+}\boldsymbol{\Psi}^{\top}(\boldsymbol{\Psi}\boldsymbol{\Psi}^{\top})^{-1}$ 是底层无噪声系统的前向 Koopman 算子。现在，利用矩阵扰动理论，在假设4下，$\boldsymbol{T_3}$ 是两个小量的乘积，因此可以忽略，从而得到 $||\boldsymbol{T_3}|| \ll ||\boldsymbol{T_1}||$。因此，忽略 $\boldsymbol{T_3}$，

$$\boldsymbol{K_{fm}}\approx\boldsymbol{K_f}(\boldsymbol{I}-\boldsymbol{P_1})+\boldsymbol{T_2}=\boldsymbol{K_f}\boldsymbol{\mathcal{B}}+\boldsymbol{\Delta_f},\tag{18}$$

其中 $\boldsymbol{\mathcal{B}}=\boldsymbol{I}-\boldsymbol{P_1}$ 以及 $\boldsymbol{\Delta_f}=\boldsymbol{T_2}$。由于 $\boldsymbol{K_{fm}}$ 由与噪声相关的统计元素组成，我们计算期望值为 $\mathbb{E}[\boldsymbol{K_{fm}}]\approx\boldsymbol{K_f}\mathbb{E}[\boldsymbol{\mathcal{B}}]+\mathbb{E}[\boldsymbol{\Delta_f}]$，其中 $\mathbb{E}[\boldsymbol{\mathcal{B}}]=\boldsymbol{I}-\mathbb{E}[\boldsymbol{P_1}]=\boldsymbol{I}-(\mathbb{E}[\boldsymbol{N_{\psi}}]\boldsymbol{\Psi}^{\top}+\boldsymbol{\Psi}\mathbb{E}[\boldsymbol{N_{\psi}^{\top}}]+\mathbb{E}[\boldsymbol{N_{\psi}}\boldsymbol{N_{\psi}^{\top}}])(\boldsymbol{\Psi}\boldsymbol{\Psi}^{\top})^{-1}$表示由噪声引起、影响 $\boldsymbol{K_f}$ 的乘性偏差项。重要的是，不同于 DMD 在对 [18], 进行升维时所依赖的线性基函数，注意到 $\mathbb{E}[\boldsymbol{N_{\psi}}]\neq 0$，因为在本研究中，噪声是通过非线性基函数进行升维的。对于学习反向 Koopman 算子（13）的情况，也可以得到与（18）类似的结论，即

$$\boldsymbol{K_{bm}}\approx\boldsymbol{K_b}\boldsymbol{\mathcal{B}}+\boldsymbol{\Delta_b},\tag{19}$$

以及 $\mathbb{E}[\boldsymbol{K_{bm}}]\approx\boldsymbol{K_b}\mathbb{E}[\boldsymbol{\mathcal{B}}]+\mathbb{E}[\boldsymbol{\Delta_b}]$，其中 $\boldsymbol{\Delta_b}=(\boldsymbol{N_{\psi}^{-}}\boldsymbol{\Psi}^{\top}+\boldsymbol{\Psi^{-}}\boldsymbol{N_{\psi}^{\top}}+\boldsymbol{N_{\psi}^{-}}\boldsymbol{N_{\psi}^{\top}})(\boldsymbol{\Psi}\boldsymbol{\Psi}^{\top})^{-1}$。接着，与 [18],类似，由于提升空间中的噪声统计特性仍然未知，因此直接使用式（18）和（19）中 $\boldsymbol{K_{fm}}$与 $\boldsymbol{K_{bm}}$ 的表达式会更加简便。具体而言，为了抵消影响名义 Koopman 算子的偏差项$\boldsymbol{\mathcal{B}}$ 所带来的影响，提出如下抗噪（低偏差）Koopman 算子

$$\boldsymbol{K_{proposed}}=\sqrt{\boldsymbol{K_{fm}}\boldsymbol{K_{bm}^{-1}}}.\tag{20}$$

使用反向算子的逆来估计前向 Koopman 算子 $\boldsymbol{K_{fm}^{back}}=\boldsymbol{K_{bm}^{-1}}$，会导致式（19）中的偏差项被求逆。因此，式（20）中前向和后向动力学的乘积往往会抵消噪声引起的偏差中的主要部分，从而得到偏差较小的 Koopman 算子估计。不过，应注意，式（18）和（19）中分别存在项 $\boldsymbol{\Delta_f}$ 和 $\boldsymbol{\Delta_b}$，这会导致式（20）所定义的提出的算子 $\boldsymbol{K_{proposed}}$ 仍存在残余偏差。

接下来，证明提出的抗噪 Koopman 算子 $\boldsymbol{K_{proposed}}=\sqrt{\boldsymbol{K_{fm}}\boldsymbol{K_{bm}^{-1}}}$ 满足以下不等式就变得十分重要：

$$||\mathbb{E}[\boldsymbol{K_{proposed}^{2}}]-\boldsymbol{K_f^2}||<||\mathbb{E}[\boldsymbol{K_{fm}^{2}}]-\boldsymbol{K_f^2}||.\tag{21}$$

为此，利用假设4并结合式（16），可得

$$\boldsymbol{K_{bm}^{-1}}=\boldsymbol{\mathcal{B}}^{-1}\boldsymbol{K_b^{-1}}-\boldsymbol{\mathcal{B}}^{-1}\boldsymbol{K_b^{-1}}\boldsymbol{\Delta_b}\boldsymbol{\mathcal{B}}^{-1}\boldsymbol{K_b^{-1}}.\tag{22}$$

将 (18) 和 (22) 相乘，得到：

$$\begin{aligned}\boldsymbol{K_{fm}}\boldsymbol{K_{bm}^{-1}}=&\boldsymbol{K_f^2}-\boldsymbol{K_f^2}\boldsymbol{\Delta_b}\boldsymbol{\mathcal{B}}^{-1}\boldsymbol{K_b^{-1}}+\boldsymbol{\Delta_f}\boldsymbol{\mathcal{B}}^{-1}\boldsymbol{K_b^{-1}}\\&-\boldsymbol{\Delta_f}\boldsymbol{\mathcal{B}}^{-1}\boldsymbol{K_b^{-1}}\boldsymbol{\Delta_b}\boldsymbol{\mathcal{B}}^{-1}\boldsymbol{K_b^{-1}}.\end{aligned}\tag{23}$$

根据假设4并利用 (16)，可得 $\boldsymbol{\mathcal{B}}^{-1}=(\boldsymbol{I}-\boldsymbol{P_1})^{-1}\approx(\boldsymbol{I}+\boldsymbol{P_1})$。因此，由 (23) 可得：

$$\begin{aligned}\boldsymbol{K_{fm}}\boldsymbol{K_{bm}^{-1}}=&\boldsymbol{K_f^2}-\boldsymbol{K_f^2}\boldsymbol{\Delta_b}\boldsymbol{K_b^{-1}}-\boldsymbol{K_f^2}\boldsymbol{\Delta_b}\boldsymbol{P_1}\boldsymbol{K_b^{-1}}\\&+\boldsymbol{\Delta_f}\boldsymbol{K_b^{-1}}+\boldsymbol{\Delta_f}\boldsymbol{P_1}\boldsymbol{K_b^{-1}}-\boldsymbol{\Delta_f}\boldsymbol{K_b^{-1}}\boldsymbol{\Delta_b}\boldsymbol{K_b^{-1}}\\&-\boldsymbol{\Delta_f}\boldsymbol{P_1}\boldsymbol{K_b^{-1}}\boldsymbol{\Delta_b}\boldsymbol{K_b^{-1}}-\boldsymbol{\Delta_f}\boldsymbol{K_b^{-1}}\boldsymbol{\Delta_b}\boldsymbol{P_1}\boldsymbol{K_b^{-1}}\\&-\boldsymbol{\Delta_f}\boldsymbol{P_1}\boldsymbol{K_b^{-1}}\boldsymbol{\Delta_b}\boldsymbol{P_1}\boldsymbol{K_b^{-1}}.\end{aligned}\tag{24}$$

在假设4下，注意到 $||\boldsymbol{P_1}||\approx||\boldsymbol{\Delta_f}||\approx|\boldsymbol{\Delta_b}||\ll 1$。因此，(24) 中由两个或更多此类矩阵的乘积构成的高阶项变得可以忽略，于是利用 $\boldsymbol{K_f}=\boldsymbol{K_b^{-1}}$，

$$\boldsymbol{K_{fm}}\boldsymbol{K_{bm}^{-1}}-\boldsymbol{K_f^2}\approx-\boldsymbol{K_f^2}\boldsymbol{\Delta_b}\boldsymbol{K_f}+\boldsymbol{\Delta_f}\boldsymbol{K_f}=\boldsymbol{S_1}\boldsymbol{K_f},\tag{25}$$

其中 $\boldsymbol{S_1}=(\boldsymbol{N_{\psi}^{+}}-\boldsymbol{K_f^2}\boldsymbol{N_{\psi}^{-}})(\boldsymbol{\Psi}^{\top}+\boldsymbol{N_{\psi}^{\top}})(\boldsymbol{\Psi}\boldsymbol{\Psi}^{\top})^{-1}$。现在，利用 $\boldsymbol{\mathcal{B}}=\boldsymbol{I}-\boldsymbol{P_1}$ 和 (18)，

$$\begin{aligned}\boldsymbol{K_{fm}^{2}}-\boldsymbol{K_f^2}\approx&\ (\boldsymbol{K_f}\boldsymbol{\mathcal{B}}+\boldsymbol{\Delta_f})(\boldsymbol{K_f}\boldsymbol{\mathcal{B}}+\boldsymbol{\Delta_f})-\boldsymbol{K_f^2}\\=&-\boldsymbol{K_f}\boldsymbol{P_1}\boldsymbol{K_f}-\boldsymbol{K_f^2}\boldsymbol{P_1}+\boldsymbol{K_f}\boldsymbol{P_1}\boldsymbol{K_f}\boldsymbol{P_1}+\boldsymbol{K_f}\boldsymbol{\Delta_f}\\&-\boldsymbol{K_f}\boldsymbol{P_1}\boldsymbol{\Delta_f}+\boldsymbol{\Delta_f}\boldsymbol{K_f}-\boldsymbol{\Delta_f}\boldsymbol{K_f}\boldsymbol{P_1}+\boldsymbol{\Delta_f^2}.\end{aligned}\tag{26}$$

类似地，(26) 中涉及两个或更多此类矩阵乘积的高阶项变得可以忽略，因此，

$$\begin{aligned}\boldsymbol{K_{fm}^{2}}-\boldsymbol{K_f^2}\approx&(\boldsymbol{\Delta_f}-\boldsymbol{K_f}\boldsymbol{P_1})\boldsymbol{K_f}+\boldsymbol{K_f}(\boldsymbol{\Delta_f}-\boldsymbol{K_f}\boldsymbol{P_1})\\=&[(\boldsymbol{N_{\psi}^{+}}-\boldsymbol{K_f}\boldsymbol{N_{\psi}})\boldsymbol{\Psi}^{\top}+(\boldsymbol{\Psi}^{+}-\boldsymbol{K_f}\boldsymbol{\Psi})\boldsymbol{N_{\psi}^{\top}}\\&+(\boldsymbol{N_{\psi}^{+}}-\boldsymbol{K_f}\boldsymbol{N_{\psi}})(\boldsymbol{N_{\psi}^{\top}})](\boldsymbol{\Psi}\boldsymbol{\Psi}^{\top})^{-1}\boldsymbol{K_f}\\&+\boldsymbol{K_f}[(\boldsymbol{N_{\psi}^{+}}-\boldsymbol{K_f}\boldsymbol{N_{\psi}})\boldsymbol{\Psi}^{\top}+(\boldsymbol{\Psi}^{+}-\boldsymbol{K_f}\boldsymbol{\Psi})\boldsymbol{N_{\psi}^{\top}}\\&+(\boldsymbol{N_{\psi}^{+}}-\boldsymbol{K_f}\boldsymbol{N_{\psi}})(\boldsymbol{N_{\psi}^{\top}})](\boldsymbol{\Psi}\boldsymbol{\Psi}^{\top})^{-1}=\boldsymbol{S_2}\boldsymbol{K_f}+\boldsymbol{K_f}\boldsymbol{S_2},\end{aligned}\tag{27}$$

其中，我们用到了 $\boldsymbol{\Psi}^{+}=\boldsymbol{K_f}\boldsymbol{\Psi}$ 这一事实，并且 $\boldsymbol{S_2}=(\boldsymbol{N_{\psi}^{+}}-\boldsymbol{K_f}\boldsymbol{N_{\psi}})(\boldsymbol{\Psi}^{\top}+\boldsymbol{N_{\psi}^{\top}})(\boldsymbol{\Psi}\boldsymbol{\Psi}^{\top})^{-1}$。在 (25) 和 (27) 中，项 $\boldsymbol{S_1}$ 和 $\boldsymbol{S_2}$ 包含经非线性函数提升的随机噪声分量，因此无法明确刻画其统计特性。然而，我们利用离散时间 Koopman算子 $\boldsymbol{K_f}$ 可写为：

$$\boldsymbol{K_f}=e^{\boldsymbol{\mathcal{K}}\boldsymbol{\Delta t}}=\boldsymbol{I}+\underbrace{\boldsymbol{\mathcal{K}}\Delta t}_{\boldsymbol{R}}+\mathcal{O}(\Delta t^2),\tag{28}$$

其中，$\Delta t(\ll 1)$ 是采样时间，而 $\boldsymbol{\mathcal{K}}$ 是与无噪声系统 (3) 对应的（未知）理想连续时间Koopman表示。因此，当采样间隔足够小时，我们忽略 (28) 中的高阶项，并得到

$$\boldsymbol{K_f}\approx\boldsymbol{I}+\boldsymbol{R}\qquad||\boldsymbol{R}||\ll 1\qquad||\boldsymbol{N_{\psi}}||\approx||\boldsymbol{N_{\psi}^{+}}||\approx||\boldsymbol{N_{\psi}^{-}}||\text{with}。由于，很容易验证，对于某个常数\ \delta>0，有||\mathbb{E}[\boldsymbol{S_1}]||\approx||\mathbb{E}[\boldsymbol{S_2}]||\approx\delta，因此，$$

$$\mathbb{E}[\boldsymbol{K_{fm}^{2}}]-\boldsymbol{K_f^2}\approx 2\mathbb{E}[\boldsymbol{S_2}],\ \mathbb{E}[\boldsymbol{K_{fm}}\boldsymbol{K_{bm}^{-1}}]-\boldsymbol{K_f^2}\approx\mathbb{E}[\boldsymbol{S_1}].\tag{29}$$

因此，我们得到，

$$||\mathbb{E}[\boldsymbol{K^2_{proposed}}]-\boldsymbol{K_f^2}|| \approx \delta, \ ||\mathbb{E}[\boldsymbol{K^2_{fm}}]-\boldsymbol{K_f^2}||\approx 2\delta. \quad (30)$$

因此，(21) 中的不等式可直接得出，采用所提出的 Koopman 算子能够减小噪声引起的偏差。显然，同时纳入前向和后向时间动力学，可以缓解学习得到的 Koopman 算子中的噪声引起的偏差，而无需显式估计污染提升状态的噪声统计特性；这是所提出方案的一项重要特征。

注 *2*：上述理论分析将 [18], 中提出的前向-后向 DMD 框架扩展到了原本仅适用于以原始状态作为 Koopman 可观测量的自治系统的情形。相比之下，本文的分析适用于控制仿射形式的非自治系统中的非线性 Koopman 可观测量；当系统为自治系统且直接将状态视为 Koopman 可观测量时，该分析便退化为 [18] 中的表述。需要特别指出的是，综合偏差较小的 Koopman 算子依赖于不等式 (14) 成立，而在噪声水平较高时，该不等式可能无法严格成立。然而，下一节中的仿真结果清楚地表明，即使测量噪声水平相对较高，所提出的架构仍然具有鲁棒性，并能有效缓解噪声引起的偏差；这是所提出方案的一项重要贡献。

V. 仿真结果

本节通过全面的仿真研究，评估所提出的 DRKN架构从含噪数据中学习线性 Koopman 模型的有效性。我们考察了两种具有代表性的非线性系统：范德波尔振荡器和串联机械臂。首先，使用受测量噪声污染的数据集训练 DRKN，以模拟真实环境。为评估所学习模型的质量，我们在无噪声轨迹上评估其预测性能，从而将模型精度与噪声影响分离。然而，对于闭环轨迹跟踪，假设反馈带有噪声，以反映真实的传感器环境。此外，我们将 DRKN 的性能与领先的先进方法进行比较，包括 DKND [24],循环滚动预测 [25],以及名义Koopman（NLK） [26], ，结果表明所提出的方法具有更高的精度和鲁棒性。最后，还展示了所学习模型在各种机械臂系统中实现精确闭环轨迹跟踪的有效性。

A. 范德波尔振荡器

Co考虑带控制输入的范德波尔振荡器 ：

$$\dot{x}_1 = -x_2,\ \dot{x}_2 = \mu(-1+{x_1}^2)x_2 + x_1 + u \quad (31)$$

其中， u 是控制输入， $\mu = 1$。为收集数据，时间步长 Δt 设为 0.01s，控制输入 u随机采样。用于训练 DRKN 的训练超参数如表 I 所示。

表 I：Koopman网络的超参数及数据集信息（(T,S)：T——轨迹数量，S——每条轨迹中的快照数量）

	范德波尔振荡器	4R机械臂 tor	Franka FR3 机械臂
DRKN自编码器			
架构	[2, 20, 20, 20, 10]	[8, 40, 40, 40, 20]	[14, 30, 30, 30, 60]
提升状态	12	28	74
$\alpha_1, \alpha_2, \alpha_3$	1, 0.5, 0.01	1, 0.5, 0.01	1, 0.5, 0.01
γ_1, γ_2	0, 0	0, 0	0, 0
批量大小	256	256	256
优化器	adam	adam	adam
数据集	(100, 100)	(350, 350)	(820, 1000)

表 II：范德波尔振荡器的平均预测误差对比。

范德波尔振荡器				
噪声	提出的	DKND [24]	循环 滚动展开 [25]	NLK [26]
20dB	**0.097**	0.476	0.127	0.466
25dB	**0.077**	0.217	0.089	0.225
30dB	**0.068**	0.092	0.088	0.106
35dB	**0.061**	0.069	0.087	0.074
40dB	0.058	0.065	**0.054**	0.070

仿真实验在英特尔 i7-12700 处理器上进行。图 2 展示了 DRKN 的开环预测性能对应结果。预测性能通过平均预测误差量化，其定义为 $e_{pred} = (1/N_{snaps})$ $\sum_{i=1}^{N_{snaps}} ||\boldsymbol{x_{i,\ true}} - \boldsymbol{x_{i,\ pred}}||_2$其中 $\boldsymbol{x_{true}}$ 和 $\boldsymbol{x_{pred}}$ 分别为真实状态轨迹和预测状态轨迹， N_{snaps} 为预测所考虑的时间步数。如图 2b 和表 II 所示，DRKN 在预测精度方面显著优于 DKND[24] 和 NLK [26]架构。NLK 架构未考虑数据中包含的噪声，因此随着训练数据中的噪声水平增加，所预测的 Koopman 算子会变得不准确。DKND [24]方法在其损失函数中增加了一个额外项，用于惩罚从含噪数据和无噪声数据中学习到的 Koopman 算子之间的差异。然而，由于真实的 Koopman 算子仍然未知，该公式用一个过于保守的上界代替误差。因此，优化器主要通过调整基函数来最小化这个保守上界，而基本忽略了噪声本身的主导影响。因此，DKND 在减轻式（18）中训练噪声引入的内在偏差方面效果相对有限；式（18）量化了对于给定的一组基函数，噪声如何在学习到的 Koopman 算子中传播。相比之下，提出的 DRKN 方法（10）消除了噪声偏差中的这一重要部分，从而能够从含噪数据中更准确地学习 Koopman 算子。 [25] 中的方法采用 polyflow 基函数，并在训练过程中计算预测损失时逐步增加预测时域。尽管对于范德波尔振荡器，该方法取得了与 DRKN 相当的预测性能，但由于计算预测损失所需的展开长度逐步增加，其计算成本显著更高，如

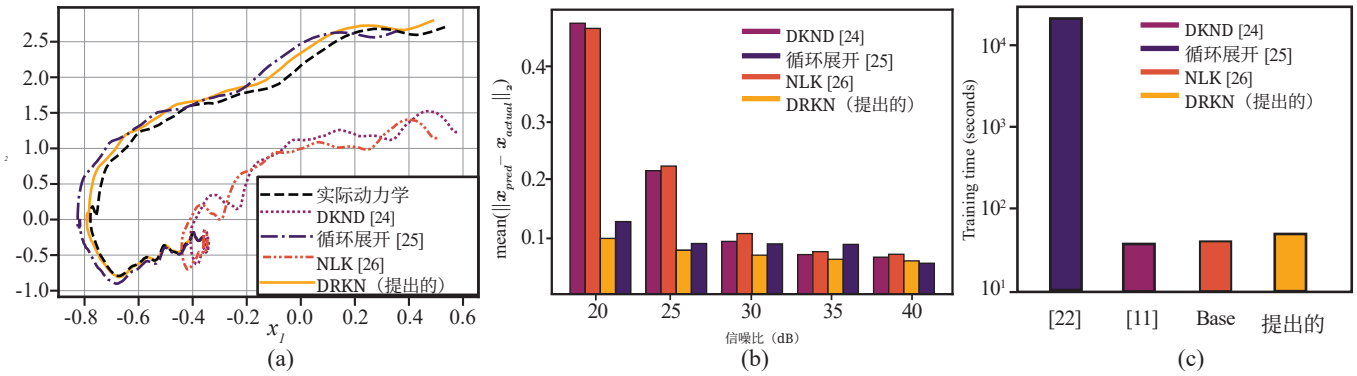

图 2. 范德波尔振荡器动力学学习的性能比较 a）使用信噪比为 20 dB 的数据，通过学习得到的Koopman模型生成的相图。b）预测性能比较。c）数据集受到 20 dB 噪声污染时的训练时间比较。

图 2c 和表 II 所示。因此，DRKN 在精度和计算效率之间实现了良好的权衡，使其成为真实场景中模型学习更实用的选择。

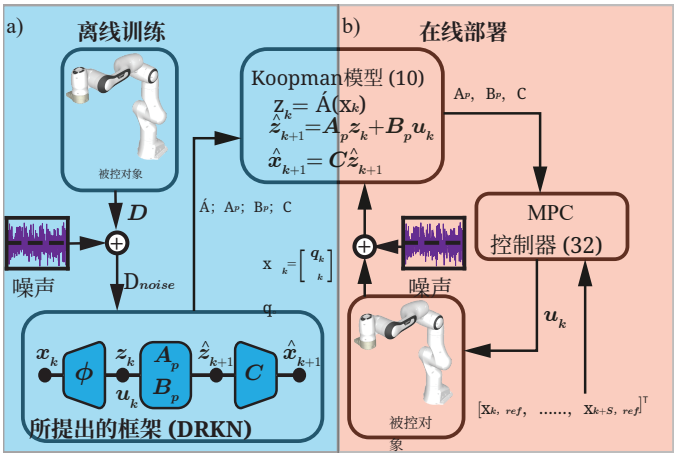

图 3. 所提出的Koopman-MPC框架示意图

B. 模型预测控制

对于后续的跟踪控制研究，如图 3 所示，提出的架构与线性模型预测控制（模型预测控制）方案相结合，利用学习得到的 Koopman模型，在关节/输入约束下实现高效且可靠的性能。对于预测时域 H，线性模型预测控制实质上求解以下优化问题，

$$\min_{Z,U} \sum_{k=0}^{H-1} \begin{bmatrix} Cz_k - x_{ref,k} \\ u_k \end{bmatrix}^T Q \begin{bmatrix} Cz_k - x_{ref,k} \\ u_k \end{bmatrix}, \quad (32)$$

满足 $z_{k+1} = A_p z_k + B_p u_k, z_0 = \phi(x_0)$

$x^- \le Cz_k \le x^+, u^- \le u_k \le u^+, k = 0, 1, \ldots, H-1$ 其中 $x_{ref,k}$ 指预测时域中第 k^{th} 个时间步的参考状态。Q 表示针对状态和输入组合的惩罚矩阵，而状态与输入边界约束分别由$[x^-, x^+]$ 和 $[u^-, u^+]$ 表示。

C. 4R机械臂仿真结果

接下来，我们在一个工作于三维工作空间的 4R 串联机械臂上评估所提出的架构，并使用 Python RoboticsToolbox [33] 实现。对于该 4R机械臂，用于数据采集的模型参数如下：对于 i^{th} 连杆，质量为 $m_i = 0.1$ kg，长度为 $l_i = 0.33$ m，质心位于 $a_i = l_i/2$，转动惯量为 $I_i = \text{diag}[0, 1.5, 1.5]$ kg·m^2。我们从预测精度和跟踪性能两方面，将所提出的方法与相关先进方法进行基准比较；后者是在模型预测控制框架 (32) 下评估的，该框架用于跟踪正弦关节空间轨迹。相应的超参数和数据集信息见表 I。

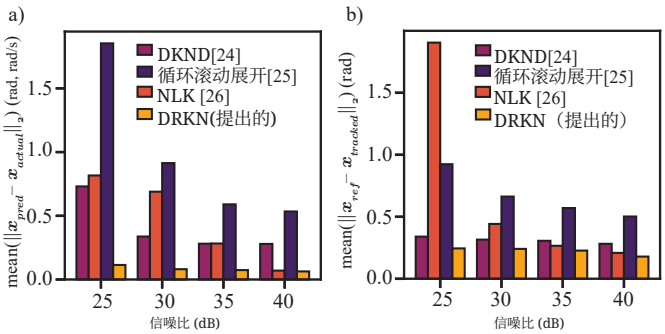

图 4. a) 4R机械臂动力学学习的预测性能对比。b) 受 30分贝噪声污染的反馈下的闭环跟踪误差对比。

4R系统的训练数据集首先被加入噪声，使其信噪比值介于 25−40dB之间。随后，使用这些含噪数据集学习所有对比方法的Koopman 算子，包括提出的DRKN方法、DKND [24], 循环展开 [25], 和NLK模型 [26]。针对真实轨迹计算相应的开环预测误差，误差结果如图4所示。显然，在预测性能方面，DRKN再次优于DKND [24], 循环展开 [25],和NLK [26]架构，展现出其准确捕捉非线性系统动力学并保持对测量噪声鲁棒性的能力——

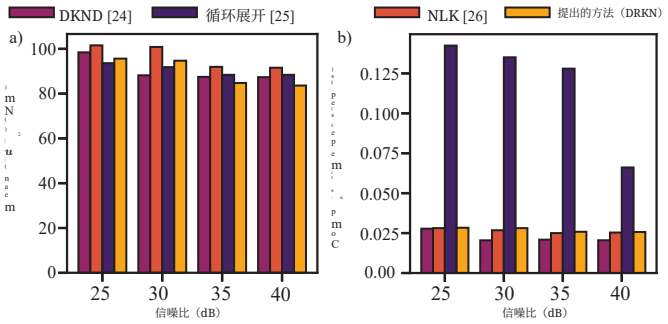

图 5. 使用在不同噪声水平下训练的 Koopman模型，在反馈测量受到 30 dB 噪声污染时，4R机械臂的 a) 控制力度和 b) MPC 每步计算时间的闭环对比。

噪声。随后，为评估跟踪控制性能，将先前学习得到的Koopman模型整合到模型预测控制（MPC）框架中，以跟踪图3所示的正弦关节空间轨迹。跟踪过程中，反馈测量值还被加入显著的高斯噪声，其信噪比为30dB。跟踪性能通过平均跟踪误差量化，其定义为 $e_{track} = (1/N) \sum_{i=1}^{N} ||\boldsymbol{\theta}_{i,\ ref} - \boldsymbol{\theta}_{i,\ track}||_2$，其中 $\boldsymbol{\theta}_{ref}$ 和 $\boldsymbol{\theta}_{track}$ 分别为参考关节角与跟踪关节角，N 为跟踪所考虑的时间步数。需要强调的是，在反馈路径中引入噪声，纯粹是为了模拟闭环跟踪期间的真实运行条件。虽然这有助于展示所学习模型在闭环控制中的有效性，但利用含噪数据开发准确模型才是本研究的主要贡献。

图 4 展示了从不同噪声水平下的训练数据集学习得到的 Koopman模型对应的关节位置跟踪误差。表 III 对预测误差和跟踪误差指标进行了定量比较。显然，提出的 DRKN 方法始终优于 DKND [24]，循环展开 [25] 和 NLK [26] 基线，展现出更优的跟踪性能，这与其更高的预测精度相符，同时控制力度和计算时间与其他方法相当，如图 5 所示。这一改进可归因于 DRKN 即使在存在噪声的情况下也能学习到更准确的系统动力学，从而直接提升闭环中模型预测控制器的效能。

D. Franka FR3 仿真结果

接下来，我们在高保真 Gazebo 环境中仿真的 7自由度 Franka FR3 机械臂上评估所提出算法的性能。用于学习的数据集由 820 条随机生成的轨迹组成，每条轨迹包含以 200 Hz 采样的 1000 个快照。为评估所提出架构的噪声鲁棒性，我们向状态（关节位置和速度）及扭矩输入测量值中注入信噪比（SNR）为 30分贝的零均值高斯噪声。随后，我们再次将所提出方法与 NLK [26],DKND [24] 和循环展开

表 III：不同训练数据集噪声水平下 4R机械臂的平均预测误差和跟踪误差对比。预测误差是相对于无噪声轨迹计算的。在闭环跟踪研究中，反馈状态测量受到 30分贝噪声的干扰。

噪声	提出的	DKND [24]	循环 滚动展开 [25]	NLK [26]
4R机械臂 – 预测误差（rad，rad/s）				
25dB	**0.114**	0.731	1.853	0.816
30dB	**0.081**	0.337	0.914	0.689
35dB	**0.074**	0.281	0.590	0.282
40dB	**0.063**	0.279	0.533	0.070
4R机械臂 – 跟踪误差（弧度）				
25dB	**0.245**	0.340	0.923	1.904
30dB	**0.241**	0.316	0.663	0.442
35dB	**0.227**	0.306	0.570	0.266
40dB	**0.178**	0.282	0.502	0.209
4R机械臂 – 控制力度（牛·米）				
25dB	95.62	98.42	**93.60**	101.54
30dB	94.69	**88.18**	91.83	100.84
35dB	**84.75**	87.47	88.40	91.94
40dB	**83.60**	87.34	88.39	91.56
4R机械臂 – 每次MPC迭代的计算时间（秒）				
25分贝	0.0283**0.0277**	0.142	0.0281	
30dB	0.0281	**0.0204**	0.135	0.0267
35dB	0.0258	**0.0208**	0.127	0.0250
40dB	0.0256	**0.0205**	0.066	0.0253

架构进行比较，并在模型预测控制框架（32）下通过力矩控制向 Franka FR3 发出指令，以评估轨迹跟踪性能。评估重点是机械臂的末端执行器，其任务是跟踪各种目标形状。如图6a和表 IV所示，在含噪数据上训练的所提出架构，相较于其他方法获得的 Koopman模型，展现出更高的跟踪精度。具体而言，采用 polyflow 基函数学习 Koopman模型的[25], 方法跟踪效果较差，而模型预测控制在若干次迭代后无法在 DKND模型下收敛。

虽然前面的比较结果清楚地证明了提出的方法对训练数据中噪声的鲁棒性，但我们还开展了其他研究，以模拟测量噪声水平增加的更现实条件。具体而言，在用于闭合控制回路的关节位置和速度反馈信号中加入了信噪比40 dB的测量噪声。需要注意的是，这些噪声水平与硬件实现中预期的噪声水平非常相近。在这种情况下， [24], [25] 中的方法经过若干次迭代后，模型预测控制无法收敛。如图6b和表 IV所示，采用提出的（DRKN）方法时，末端执行器仍能继续跟踪目标形状，尽管跟踪性能略有下降；鉴于反馈测量中的噪声水平更高，这种下降是预期之中的。因此，提出的DRKN框架在噪声条件下显然能够提供鲁棒且可靠的控制性能，在预测和闭环跟踪任务中均优于现有方法。

VI. 实验结果

本节评估所提出的方法在真实环境中对 Franka FR3 机械臂的跟踪性能。相同的 Koopman模型在

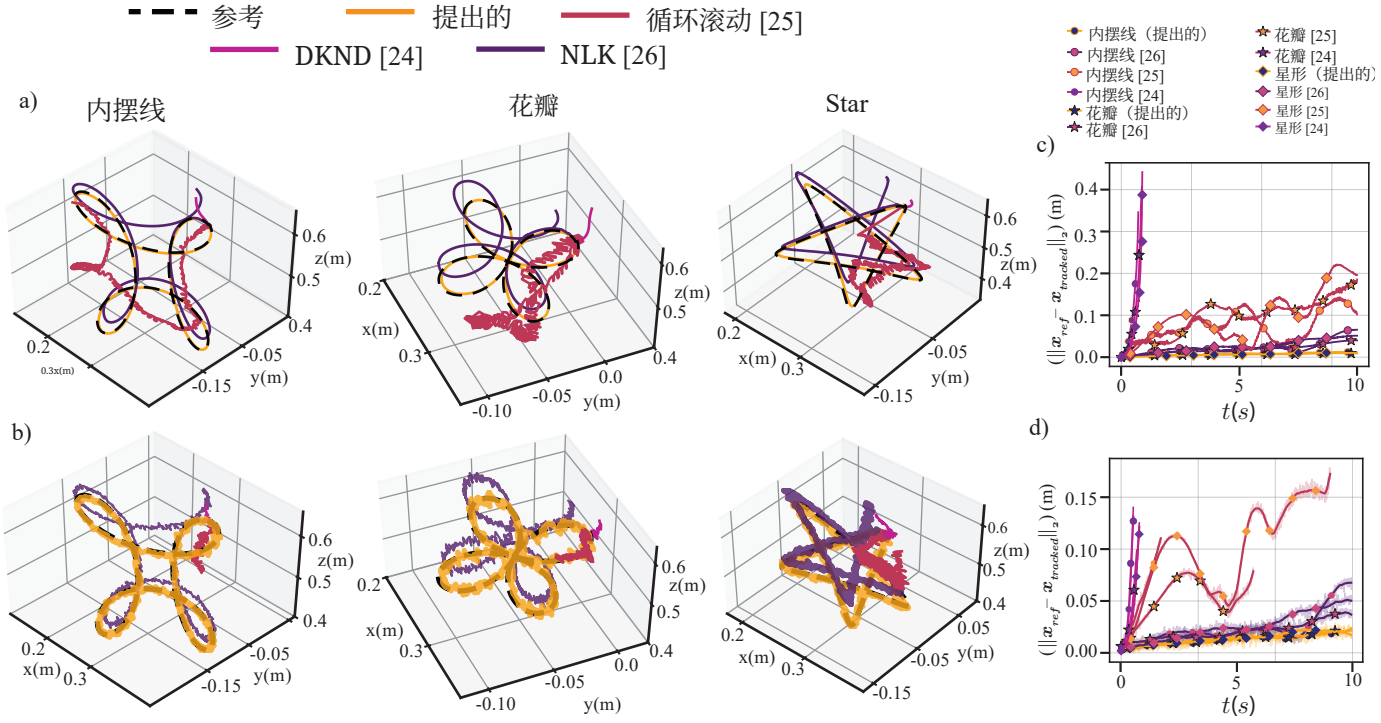

图 6. 使用在含噪数据（信噪比 = 30 dB）上训练的不同方案和 Koopman模型，在 Franka FR3 机械臂上进行轨迹跟踪。（a）低（~80 dB）和（b）高（40 dB）测量噪声下的跟踪性能，以及（c）低（~80 dB）和（d）高（40 dB）测量噪声下相应的跟踪误差变化。

表 IV：使用信噪比为 30dB 的仿真数据集在 Franka FR3 机械臂上训练的各种方案的性能比较（平均跟踪误差（m）、控制力度（N-m）和每次迭代计算时间（秒））。

	Gazebo仿真 案例 1：低反馈噪声（80 dB）			Gazebo仿真 情况 2：高反馈噪声（40 分贝）			真实世界实验 高反馈噪声 - 40分贝	
方法	假设- 摆线	花瓣	Star	内摆- 摆线	花瓣	Star	内摆- 旋轮线	三尖瓣
[24]	-	-	-	-	-	-	-	-
[25]	0.074 (22.12) (0.0046)	0.0963 (19.01) (0.0045)	0.0962 (11.57) (0.0038)	-	-	-	-	-
[26]	0.0231 **(0.94)** (0.0031)	0.0188 **(0.60)** (0.0034)	0.0254 **(0.66)** **(0.0029)**	0.023 **(1.03)** **(0.0035)**	0.021 **(0.70)** **(0.0033)**	0.025 **(0.78)** **(0.0034)**	-	-
提出的	**0.0075** (1.05) **(0.0031)**	**0.0063** (0.65) **(0.0030)**	**0.0064** (0.70) (0.0030)	**0.012** (1.44) (0.0042)	**0.013** (1.12) (0.0042)	**0.014** (1.21) (0.0043)	**0.014** **(2.43)** **(0.0099)**	**0.011** **(2.32)** **(0.0098)**

上一节所述的 Gazebo仿真数据上训练后，现已部署于模型预测控制框架（32）中。为缓解使用仿真数据集训练所造成的显著仿真到现实差距，现将广义扩展状态观测器（GESO） [34] 集成到模型预测控制框架中，以抑制未建模动态带来的扰动。需要指出的是，为确保比较公平，所有参与比较的 Koopman模型均集成在 MPC-GESO 控制器框架中。如图 7 所示，所提出的方案成功跟踪了期望轨迹，在实验涉及的两种目标形状下均实现了较低的跟踪误差。如表 IV 所示，NLK [26], DKND [24]和 Recurrent-Rollout [25]架构未能完成任务。对于 [24], [25], 中的方法，MPC-GESO 控制器未能收敛，原因是

相应 Koopman模型的仿真到现实差距非常大。此外，信噪比为 30分贝的测量噪声会导致模型 [26], 产生显著偏差，使 MPC-GESO 控制器发出不兼容的力矩和加加速度（力矩导数）指令（图 7d）。图 7c-d 中观察到的控制力度增加和力矩变化率骤变，直接源于在含噪训练条件下学习到的 Koopman模型质量不佳。这些高导数力矩输入会使 Franka 机械臂产生显著的加加速度，最终超过允许阈值，导致 Franka 控制接口中止任务。在使用含有 40分贝噪声的数据训练 Koopman模型时，也观察到了类似趋势，但为简洁起见，此处不予展示。因此，在真实环境中对 Franka FR3 机械臂进行的这项实验评估表明

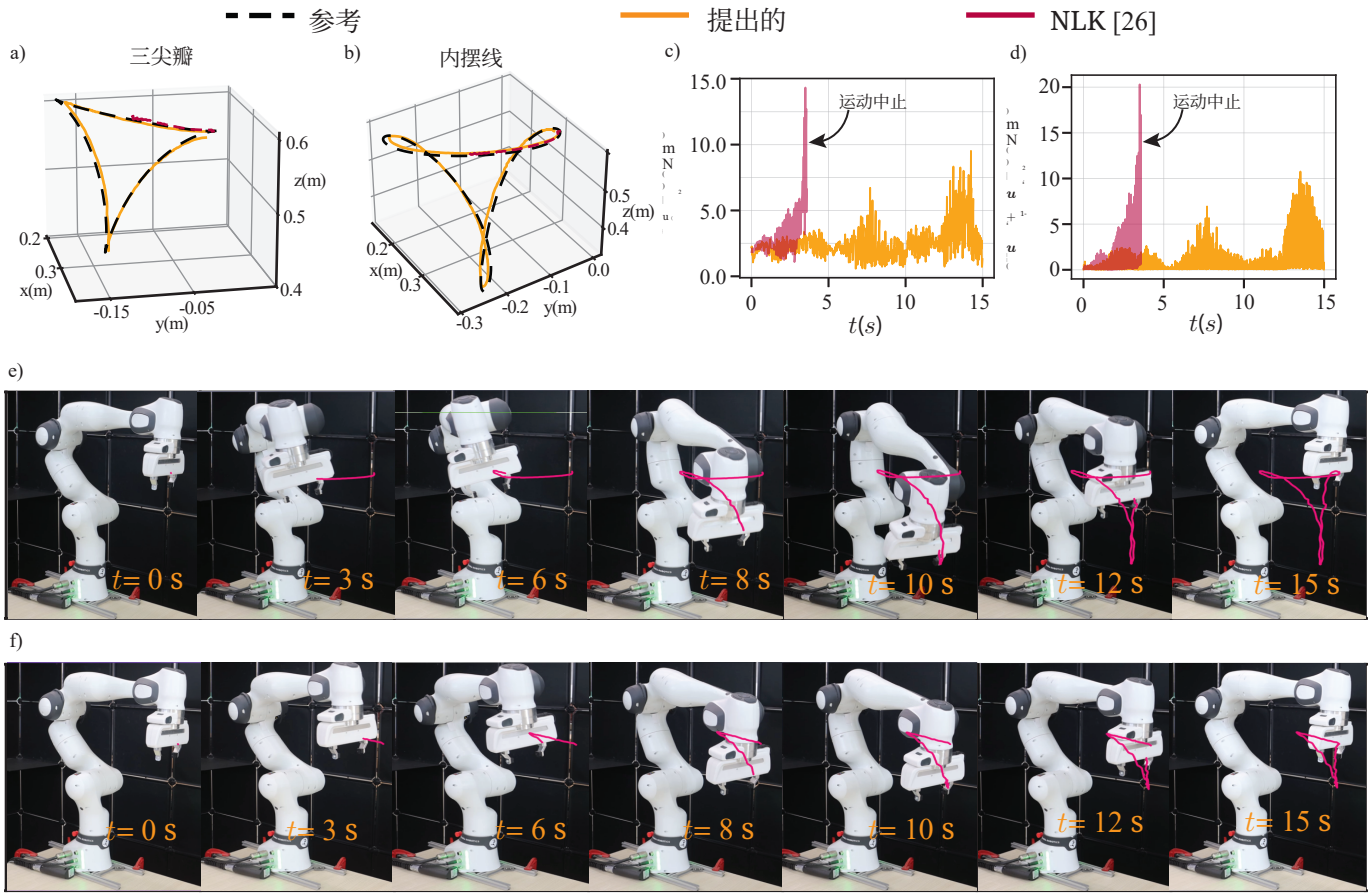

图 7. 使用由含有 30分贝信噪比噪声的数据训练得到的 Koopman模型，通过 Franka FR3 机械臂进行真实环境中的轨迹跟踪，跟踪对象为 (a,f) 三尖瓣轨迹和 (b,e) 内摆线轨迹。(c)、(d) MPC-GESO框架针对 Adobe轨迹生成的控制力矩和力矩变化率曲线，采用提出的 (DRKN) 和 NLK [26] Koopman模型。未展示其他架构（DKND [24] 和 Recurrent-Rollout [25]），因为它们会立即导致系统不稳定。

直接从含噪数据中学习 Koopman模型至关重要，尤其是在系统受到扭矩指令可行性等实际约束时。更优的跟踪性能和成功的任务执行证明，所提出的框架能够在噪声条件下准确捕获 Koopman表示，从而提升其对实时控制的适用性。

VII. 结论

本研究提出了一种深度鲁棒 Koopman 网络（DRKN），用于从含噪数据中学习精确的 Koopman模型。所提出的架构同时学习正向和反向时间动力学，以减轻训练数据中的噪声所引入的偏差。我们通过详尽的理论分析，为采用所提出方案获得的 Koopman 算子提供鲁棒的预测性能保证。通过对多个机械臂系统开展开环预测性能和闭环轨迹跟踪仿真研究，证明了所提出框架的有效性。进一步通过详细的性能比较，证明了与领先的替代设计相比，所提出方案在学习抗噪 Koopman 算子方面的有效性。最后，在真实环境中开展的实验验证研究明确凸显了合成能够在现实约束下运行的动力系统的抗噪 Koopman模型的重要性。未来的工作将利用真实世界数据扩展所提出的方案，并在线更新 Koopman 算子，以提升任务特定泛化能力。

参考文献

[1] B. O. Koopman，《哈密顿系统与希尔伯特空间中的变换》，《美国国家科学院院刊》，第17卷，第5期，第315–318页，1931年。[2] M. Korda 和 I. Mezi´c，《非线性动力系统的线性预测器：Koopman 算子与模型预测控制相遇》，*Automatica*，第93卷，第149–160页，2018年。[3] A. Mauroy、I. Mezic 和 Y. Susuki，《系统与控制中的Koopman算子：概念、方法与应用》。2020年1月。[4] J. H. Tu，动态模态分解：理论与应用。博士论文，普林斯顿大学，2013年。[5] J. L. Proctor、S. L. Brunton 和 J. N. Kutz，《带控制的动态模态分解》，*SIAM*应用动力系统期刊，第15卷，第1期，第142–161页，2016年。[6] M. O. Williams、I. G. Kevrekidis 和 C. W. Rowley，《Koopman 算子的一个数据驱动近似：扩展动态模态分解》，《非线性科学杂志》，第25卷，1346，2015年，第1307页。[7] S. L. Brunton、J. L. Proctor 和 J. N. Kutz，《通过非线性动力系统稀疏辨识，从数据中发现控制方程》，《美国国家科学院院刊》，第113卷，第15期，第3932–3937页，2016年。

[8] C. K. Sah、R. Singh 和 J. Keshavan，《机器人系统辨识与控制的数据驱动范式概述》，印度科学研究所期刊，第1–34页，2025年。[9] Q. Li、F. Dietrich、E. M. Bollt 和 I. G. Kevrekidis，《结合字典学习的扩展动态模态分解：Koopman 算子的自适应数据驱动谱分解》，《混沌：非线性科学跨学科期刊》，第27卷，第10期，文章编号103111，2017年。[10] K. Champion、B. Lusch、J. N. Kutz 和 S. L. Brunton，《坐标与控制方程的数据驱动发现》，《美国国家科学院院刊》，第116卷，第45期，第22445–22451页，2019年。[11] B. Lusch、J. N. Kutz 和 S. L. Brunton，《用于非线性动力学通用线性嵌入的深度学习》，《自然·通讯》，第9卷，第1期，文章编号4950，2018年。[12] C. K. Sah、R. Singh 和 J. Keshavan，《基于Koopman零化神经网络框架的冗余机械臂实时约束跟踪控制》，《IEEE机器人与自动化快报》，2024年。[13] C. Folkestad、D. Pastor 和 J. W. Burdick，《非线性机器人动力学的情节式Koopman学习及其在快速多旋翼着陆中的应用》，载于2020年IEEE 国际机器人与自动化会议（ICRA），第9216–9222页，2020年。[14] C. Folkestad、S. X. Wei 和 J. W. Burdick，《Koopnet：Koopman双线性模型与函数字典的联合学习及其在四旋翼轨迹跟踪中的应用》，载于2022年国际机器人与自动化会议（ICRA），第1344–1350页，IEEE，2022年。[15] D. Bruder、C. D. Remy 和 R. Vasudevan，《利用Koopman 算子理论辨识软体机器人动力系统》，载于2019年国际机器人与自动化会议（ICRA），第6244–6250页，IEEE，2019年。[16] D. A. Haggerty、M. J. Banks、P. C. Curtis、I. Mezi´c 和 E. W. Hawkes，《通过Koopman算子对螺旋驱动惯性软体机械臂进行建模、降阶与控制》，arXiv 预印本 arXiv:2011.07939，2020年。[17] M. L. Casta˜no、A. Hess、G. Mamakoukas、T. Gao、T. Murphey 和 X. Tan，《基于Koopman 算子的软体游动机器人面向控制的建模》，载于2020年IEEE/ASME先进智能机电国际会议（AIM），第1679–1685页，IEEE，2020年。[18] S. T. M. Dawson、M. S. Hemati、M. O. Williams 和 C. W. Rowley，《表征并校正传感器噪声对动态模态分解的影响》，《流体实验》，第57卷，2014年。[19] L. Shi、M. Haseli、G. Mamakoukas、D. Bruder、I. Abraham、T. Murphey、J. Cortes 和 K. Karydis，《机器人学习中的Koopman 算子》，arXiv，第abs/2408.04200卷，2024年。[20] O. Sedehi、M. Yadav、M. Stender 和 S. Oberst，《利用截断储备池计算对非线性动力学进行去噪与重建》，《混沌：非线性科学跨学科期刊》，第35卷，2025年9月。[21] A. Widmann、E. Schr¨oger 和 B. Maess，《电生理数据的数字滤波器设计——实用方法》，《神经科学方法杂志》，第250卷，第34–46页，2015年。[22] I. Reinten、I. de Ronde-Brons、R. Houben 和 W. Dreschler，《个人听者对单麦克风降噪强度的偏好：噪声容忍度与信号失真容忍度之间的权衡》，《听力趋势》，第27卷，文章编号23312165231192304，2023年。[23] L. Lortie、S. Dahdah 和 J. R. Forbes，《带渐近稳定性约束的前向-后向扩展动态模态分解》，arXiv 预印本 arXiv:2403.10623，2024年。[24] W. Hao、D. Upadhyay 和 S. Mou，《利用含噪数据进行深度Koopman学习》，《机器学习研究汇刊》，第2025卷，2024年。[25] S. A. Sakib 和 S. Pan，《利用Hankel动态模态分解学习用于控制的抗噪稳定Koopman算子》，《IEEE控制系统技术汇刊》，2024年。[26] R. Singh、C. K. Sah 和 J. Keshavan，《用于非线性动力系统鲁棒控制的自适应Koopman嵌入》，《国际机器人研究杂志》，第02783649251341907页，2024年。[27] O. Azencot、N. B. Erichson、V. Lin 和 M. Mahoney，《使用一致性Koopman自编码器预测序列数据》，载于《第37届国际机器学习会议论文集》（H. D. III 和 A. Singh 编），第119卷，《机器学习研究论文集》，第475–485页，PMLR，2020年7月13–18日。[28] D. Goswami 和 D. A. Paley，《控制仿射非线性系统的全局双线性化与可控性：一种Koopman谱方法》，载于2017年IEEE第56届决策与控制年会（CDC），第6107–6112页，IEEE，2017年。[29] M. Haseli 和 J. Cort´es，《从数据中学习Koopman特征函数与不变子空间：对称子空间分解》，《IEEE自动控制汇刊》，第67卷，第7期，第3442–3457页，2022年。[30] S. E. Otto 和 C. W. Rowley，《用于动力系统估计与控制的 Koopman 算子》，《控制、机器人与自主系统年度评论》，第 4 卷，第 4 卷，2021 年，第 59–87 页，2021 年。[31] C. Szegedy、W. Zaremba、I. Sutskever、J. Bruna、D. Erhan、I. Goodfellow 和 R. Fergus，《神经网络的奇妙特性》，*arXiv* 预印本 *arXiv:1312.6199*，2014 年。[32] G. H. Golub 和 C. F. van Loan，《矩阵计算》。约翰斯·霍普金斯大学出版社，第 4 版，2013 年。[33] P. Corke 和 J. Haviland，《这不是你奶奶的工具箱——为 Python 重塑机器人学工具箱》，载于*2021 IEEE* 国际机器人与自动化会议（*ICRA*），第 11357–11363 页，2021 年。[34] R. Singh、A. Singh、C. S. Kashyap 和 J. Keshavan，《用于鲁棒控制欧拉-拉格朗日系统的广义基于动量的 Koopman 形式体系》，*arXiv* 预印本 *arXiv:2509.17010*，2025 年。